B8IT120 – Final Project

**Name:**  Cathal Peelo

**Student Number:** 11332506

Contents

[Topic and Objectives 3](#_Toc499320790)

[Target Audience 3](#_Toc499320791)

[Rationale behind Development Approach 3](#_Toc499320792)

[Cloud Services and Technologies 4](#_Toc499320793)

[Room for Improvement 4](#_Toc499320794)

[Use Cases 4](#_Toc499320795)

[UC1: Register 4](#_Toc499320796)

[UC2: Log in 5](#_Toc499320797)

[UC3: View real-time Dublin Bikes information 5](#_Toc499320798)

[UC4: Create new journey record 6](#_Toc499320799)

[UC5: View journey records 7](#_Toc499320800)

[Bibliography 8](#_Toc499320801)

[Appendix: Project Log 9](#_Toc499320802)

# Topic and Objectives

Any resident of any city in the world has inevitably suffered the frustration that is a traffic jam. Getting from one side of the city to the other in a quick and efficient manner rather than sitting in gridlock can mean the difference between arriving at work well-rested and on time, and showing up sleep-deprived and dishevelled from spending an interminable time in the sweatbox that is the average bus or train at rush hour. Meanwhile, public health continues to deteriorate with 19.5% of adults in OECD countries classified as obese, a number which increases to over 50% when overweight people are added[[1]](#footnote-1). Traffic and the stress it results in are the bane of the modern commuter and making people unhealthy to boot.

An elegant solution is travelling by bike—burning calories while able to move through traffic or avoiding it altogether—but many people do not want to shell out for an expensive bike or risk it being robbed. In Dublin, a public bike sharing system exists but is under-used as many people are still only vaguely aware of its existence, never mind where the nearest bike station to them is or whether there are any bikes available from it at a given time. This lack of communication means Dublin Bikes is not having the positive health impact it could, a position which could easily be changed with the release of a new app which would allow users to keep track of their bike journeys and by proxy the progress they are making with their health.

# Target Audience

Although this app hopes to reduce the prevalence of obesity in the general population of Dublin by promoting exercise in the city, its target audience is not limited to that demographic. It hopes to have appeal throughout the population of Dublin—and even among those just visiting—and its simple design and relatively horizontal navigation structure should be easily usable by young and old alike.

In order to make the app as accessible as possible, it was developed using a minimum SDK version of 17 meaning that anyone using a device with android 4.2 or higher will be able to use it. Since an estimated 93.8% of android users use Jelly Bean 4.2 or higher as of September 2017[[2]](#footnote-2), this should serve to maximise access to access to this information.

# Rationale behind Development Approach

It was decided to develop a native Android app rather than go for a hybrid approach to avoid the danger of running into out-dated plugins on a device and therefore losing access to the device’s APIs[[3]](#footnote-3). Since features such as GPS and the camera are vital for several of the app’s functionalities, that could ruin its usability. On the other hand, speed to market is not a major consideration given that Dublin Bikes has been active for years and they do not expect apps to provide a major revenue stream. This means the speed benefit of using a hybrid approach was made trivial.

# Cloud Services and Technologies

Firebase cloud services is used throughout the app to provide authentication of user profiles and login. This means that a user can log in on any Android device they like, and once a user has logged in on their device they will stay logged, not needing to go through the Loading and Login Activities until they log out again.

A Firebase database is also used to store the user’s journeys. The app creates Journey objects based on the user’s specifications before saving them to the Firebase database which converts the Journey’s data members into nodes in its tree. Although it is not apparent from its online documentation, Firebase does not accept Bitmap images, so all photos are converted into strings before being uploaded.

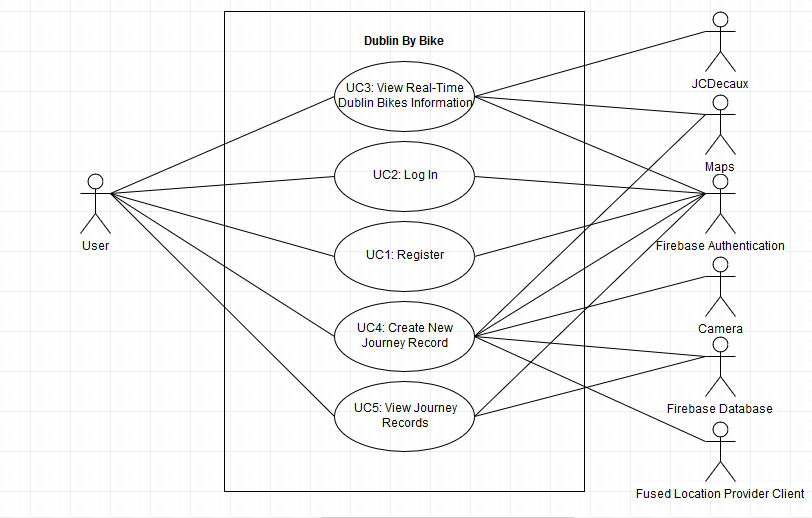
The app creates a HTTP URL Connection to request JSON data from an online API using GET. The app then parses this JSON data into JSON objects and strings. Geolocation services are provided by Android’s API Fused Location Provider Client. This provides a fine-scale Location object from which the app derives latitude and longitude. The device’s default Camera and Maps apps are used when needed.

# Room for Improvement

* The app’s no-frills, stripped-back aesthetic may underwhelm users who are used to a flashier user interface
* Could simplify Add Journey Activity by removing latitude and longitude EditTexts.
* Could add a listener in the Journey History Activity to view the downloaded picture or show the start or end point on a map, and give users full CRUD control (create, read, update and delete)[[4]](#footnote-4) over their journeys
* Could add a “nearest station” functionality to API Activity
* Should put all string files in strings.xml and refer to them by ID to facilitate localisation

# Use Cases

The following is a comprehension collection of use cases for the app, prepared in accordance with UML[[5]](#footnote-5). These helped to frame the development of the app, allowing it to be as user-centred as possible.



### UC1: Register

Actors: User, Firebase Authentication

Basic Flow:

1. App opens and immediately checks Firebase Authentication
2. Firebase authentication indicates that User is not logged in
3. User proceeds to a Log In view and on to a Register view
4. User enters their details and instructs the app to proceed
5. Firebase Authentication creates a new user with email and password using the entered details
6. App redirects to Main view

Alternative Flows:

5a. User entered invalid details

1. App instructs User to re-enter the details
2. User continues **UC1** from 4

### UC2: Log in

Actors: User, Firebase Authentication

Basic Flow:

1. App opens and immediately checks Firebase Authentication
2. Firebase authentication indicates that User is not logged in
3. User proceeds to a Log In view
4. User enters details and instructs the app to proceed
5. Firebase Authentication verifies that the details entered match those of a user stored on the cloud
6. App redirects to Main view

Alternative Flows:

5a. Firebase Authentication reports that the details entered do not match those of any registered user

1. App instructs user to enter correct details
2. User continues **UC2** from 4

### UC3: View real-time Dublin Bikes information

Actors: User, Firebase Authentication, JCDecaux, Maps

Basic Flow:

1. App opens and immediately checks Firebase Authentication
2. Firebase Authentication indicates that User is logged in
3. App immediately redirects to Main view
4. User selects option to view the real-time data
5. App redirects to API view and requests real-time data from JCDecaux
6. JCDecaux returns the data
7. App parses and displays the real-time data

Alternative Flows:

2a. Firebase Authentication reports that User is not logged in

1. User performs **UC2: Log In**, then continues **UC3** from 4

6a. JCDecaux does not respond

1. App indicates that something went wrong and no data is displayed
2. User refreshes the display
3. App continues **UC3** from 5

7a. User wants more recent data

1. User refreshes the display
2. App continues **UC3** from 5

7b. User wants to view one of the stations in Maps

1. User taps on the relevant station
2. App pauses and passes the station’s co-ordinates to Maps application
3. Maps opens, displaying the station’s location

### UC4: Create new journey record

Actors: User, Firebase Authentication, Firebase Database, Fused Location Provider Client, Camera, Maps

Basic Flow:

1. App opens and immediately checks Firebase Authentication
2. Firebase Authentication indicates that User is logged in
3. App immediately redirects to Main view
4. User selects option to create new journey record
5. App redirects to Add Journey view
6. User enters and confirms journey details
7. App sends the details to be stored in Firebase Database
8. Firebase Database saves the journey

Alternative Flows:

2a. Firebase Authentication reports that User is not logged in

1. User performs **UC2: Log In**, then continues **UC4** from 4

6a. User wants to use their current location for the start or end point

1. User selects option to use geolocation
2. App requests last known location from Android’s Fused Location Provider Client
3. Fused Location Provider Client returns location

3a. App does not have permission to access location data

1. App requests location permissions from User

2. User grants permissions

3. App continues **UC4** from 6a2

1. App fills in latitude and longitude
2. User confirms
3. App continues **UC4** from 7

6b. User wants to add a photo to the journey

1. User selects option to add a photo
2. App hands over to Camera
3. User takes photo
4. Camera passes the photo back to App
5. User confirms
6. App continues **UC4** from 7

6c. User wants to check entered co-ordinates

1. User selects option to view co-ordinates in Maps
2. App passes the co-ordinates to Maps
3. Maps displays a pin at those co-ordinates
4. User returns to App using the back button and confirms
5. App continues **UC4** from 7

### UC5: View journey records

Actors: User, Firebase Authentication, Firebase Database

Basic Flow:

1. App opens and immediately checks Firebase Authentication
2. Firebase Authentication indicates that User is logged in
3. App immediately redirects to Main view
4. User selects option to view journey records
5. App redirects to Journey History view
6. App requests journeys from Firebase Database
7. Firebase Database returns journeys
8. App displays journey details

Alternative Flows:

2a. Firebase Authentication reports that User is not logged in

1. User performs **UC2: Log In**, then continues **UC5** from 4

8a. User wants more recent data

1. User refreshes the display
2. App continues **UC5** from 6

# Bibliography

Abed, R. (2016). *Hybrid vs Native Mobile Apps – The Answer is Clear*. Retrieved from <https://ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear/>

Jacobson, I., Christerson, M., Jonsson, P., Övergaard, G. (1992). *Object-Oriented Software Engineering - A Use Case Driven Approach*. Addison-Wesley.

Kilov, H. (1990). From semantic to object-oriented data modelling. *ISCI '90 Proceedings of the first international conference on systems integration on Systems integration '90*, 385-393

OECD. (2017). *Obesity Update 2017*. OECD Publishing. Retrieved from <http://www.oecd.org/health/health-systems/Obesity-Update-2017.pdf>

Statista. (2017). *Android version market share distribution among smartphone owners as of September 2017*. Retrieved from <https://www.statista.com/statistics/271774/share-of-android-platforms-on-mobile-devices-with-android-os/>

# Appendix: Project Log

The version control system used during the development of this app is GitHub. The source code can be downloaded from <https://github.com/isitaboutabicycle/AndroidCA1> . The following are the details of all commits performed on the project.

Commits on Nov 24, 2017

@isitaboutabicycle

Cosmetic changes

isitaboutabicycle committed a minute ago

Commits on Nov 22, 2017

@isitaboutabicycle

Geolocation finally works. Also removed lat and lng from API Activity…

isitaboutabicycle committed 2 days ago

Commits on Nov 17, 2017

@isitaboutabicycle

Finally got Google Play Location services imported without Gradle goi…

isitaboutabicycle committed 7 days ago

Commits on Nov 7, 2017

@isitaboutabicycle

API Activity and Journey History Activity now update the display auto…

isitaboutabicycle committed 17 days ago

@isitaboutabicycle

API Activity's listener launches Google Maps, placing a pin with the …

isitaboutabicycle committed 17 days ago

@isitaboutabicycle

API Activity now has OnItemClickListener which parses JSON object for…

isitaboutabicycle committed 18 days ago

Commits on Nov 6, 2017

@isitaboutabicycle

Correction: Journey History works

isitaboutabicycle committed 19 days ago

@isitaboutabicycle

Add Journey now works!

isitaboutabicycle committed 19 days ago

Commits on Nov 2, 2017

@isitaboutabicycle

Only minor tweaks

isitaboutabicycle committed 22 days ago

Commits on Nov 1, 2017

@isitaboutabicycle

Journey History now displays hard-coded data correctly. Cannot retrie…

isitaboutabicycle committed 23 days ago

Commits on Oct 31, 2017

@isitaboutabicycle

New Journey starts Camera, accepts the Bitmap and saves the Journey t…

isitaboutabicycle committed 25 days ago

Commits on Oct 23, 2017

@isitaboutabicycle

Add Journey works except for getting pics from the gallery

isitaboutabicycle committed on Oct 23

Commits on Oct 22, 2017

@isitaboutabicycle

Auth finally works!!! Cosmetic updates too

isitaboutabicycle committed on Oct 22

@isitaboutabicycle

Merge pull request #4 from isitaboutabicycle/JsonObjects

isitaboutabicycle committed on Oct 22

@isitaboutabicycle

API Activity finished!

isitaboutabicycle committed on Oct 22

@isitaboutabicycle

Parsed objects now print to the ListView!

isitaboutabicycle committed on Oct 22

Commits on Oct 21, 2017

@isitaboutabicycle

Code added to parse JSON objects

isitaboutabicycle committed on Oct 21

@isitaboutabicycle

Starting branch for parsing JSON to objects

isitaboutabicycle committed on Oct 21

@isitaboutabicycle

Minor format adjustment

isitaboutabicycle committed on Oct 21

@isitaboutabicycle

Merge pull request #3 from isitaboutabicycle/firebaseTake1

isitaboutabicycle committed on Oct 21

Commits on Oct 20, 2017

@isitaboutabicycle

API Activity now dumps JSON to a TextView. Does not work on emulators…

isitaboutabicycle committed on Oct 20

@isitaboutabicycle

Merge pull request #2 from isitaboutabicycle/firebaseTake1

isitaboutabicycle committed on Oct 20

Commits on Oct 10, 2017

@isitaboutabicycle

Main activity fixed. Reg and login are still not functional.

isitaboutabicycle committed on Oct 10

Commits on Oct 2, 2017

@isitaboutabicycle

Create Journey take 1 theoretically working

isitaboutabicycle committed on Oct 2

Commits on Sep 29, 2017

@isitaboutabicycle

Added a new journey activity. Nothing works but at least the main act…

isitaboutabicycle committed on Sep 29

Commits on Sep 27, 2017

@isitaboutabicycle

register activity doesn't work but it's a bit fancier

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Added a register activity

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Enabled authentication

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Started adding firebase, sorted gradle errors

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Added an empty journey history actvity

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Merge pull request #1 from isitaboutabicycle/nofrags

isitaboutabicycle committed on Sep 27

@isitaboutabicycle

Added a journey class. API still doesn't work

isitaboutabicycle committed on Sep 27

Commits on Sep 26, 2017

@isitaboutabicycle

Added API activity which builds but doesn't get the data

isitaboutabicycle committed on Sep 26

Commits on Sep 25, 2017

@isitaboutabicycle

Basic main activity working

isitaboutabicycle committed on Sep 25

@isitaboutabicycle

ditching fragments and tabs for activities

isitaboutabicycle committed on Sep 25

@isitaboutabicycle

Added fragment for journey history

isitaboutabicycle committed on Sep 25

Commits on Sep 25, 2017

@isitaboutabicycle

Added a login screen activity and a main activity

isitaboutabicycle committed on Sep 25

@isitaboutabicycle

Created welcome screen activity

isitaboutabicycle committed on Sep 25

@isitaboutabicycle

Initial commit

isitaboutabicycle committed on Sep 25

1. OECD. (2017). Obesity Update 2017. OECD Publishing. Retrieved from http://www.oecd.org/health/health-systems/Obesity-Update-2017.pdf [↑](#footnote-ref-1)
2. Statista. (2017). Android version market share distribution among smartphone owners as of September 2017. Retrieved from https://www.statista.com/statistics/271774/share-of-android-platforms-on-mobile-devices-with-android-os/ [↑](#footnote-ref-2)
3. Abed, R. (2016). Hybrid vs Native Mobile Apps – The Answer is Clear. Retrieved from https://ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear/ [↑](#footnote-ref-3)
4. Kilov, H. (1990). From semantic to object-oriented data modelling. *ISCI '90 Proceedings of the first international conference on systems integration on Systems integration '90*, 385-393 [↑](#footnote-ref-4)
5. Jacobson, I., Christerson, M., Jonsson, P., Övergaard, G. (1992). *Object-Oriented Software Engineering - A Use Case Driven Approach*. Addison-Wesley. [↑](#footnote-ref-5)